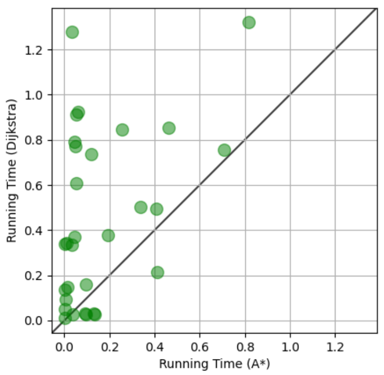

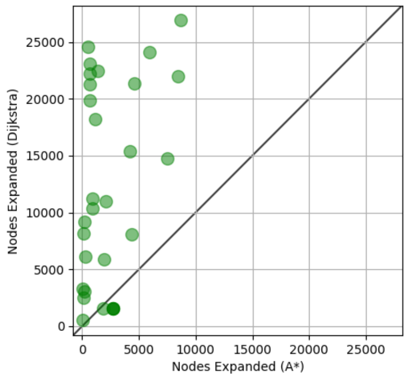

1a) To define the main line, it represent when both A* and Dijkastra perform the same in terms of run time and node expansion.

In the Expanded node graph, the majority of the dots are above the main diagonal line. This means that the A* expands a lot fewer noodles in comparison to Dijkstra. This is because of the heuristic function that A* uses to guide the path towards the goal instead of dijkstra that equally iterates through all the children of the node. Points above the main line mean that A* is using less nodes than dijkstra to get to the goal.

In the run time graph, points are mainly clustered near the bottom but mostly still above the main line. This overall still represents that A* is more efficient in terms of computation, obviously because it is not iterating through as many points as dijkstra is and A* has a shorter run time.

b) The points do not align in the same exact location in both the plots but are relatively similar because of two reasons. They are similar because you overall A* is more efficient and uses less nodes than dijkstra which results in a lower run time. A* is supposed to expand less nodes than dijkstra but the exact number of nodes vary depending on the implementation of the code especially where you call reheapify or if you add more nodes to the open list rather than update. And run time depends on a few things like the number of nodes being expanded, for example if it's more, then runtime will be more or even things like the machine you're using can make a difference in run time. So due to these two reasons the points do not align on both graphs but relatively, since a* is meant to be more efficient, the dots should be above the main diagonal line for both graphs which is the case for us.

2) Weighted A* with the weight of 2 should favour short paths over longer ones especially if normal A* which is basically a weight of 1 does. In short, WA* also finds solutions with lower costs especially relative to dijkstra. In this case the graph shows that dijkstra solutions are much cheaper than WA* solutions and therefore since their code is not running as expected, i might give them advice to check if the code is implemented correctly or if the heuristic function has an issue that can fix this issue. They can maybe make sure they are not multiplying maybe like the heuristic function by 2 but last the cost from one node to another by 2.